Comparison of SIFT & SURF Corner Detector as Features and other Machine Learning Techniques for Identification of Commonly used Leaves

Dr. Jharna Majumdar[1], Anand Mahato[2]

[1]*Dean R&D, Prof. & Head, Dept. of M.Tech CSE, Nitte Meenakshi Institute of Technology Yelahanka, Bangalore-560 064, India*
[2] *Student, M.Tech Sem IV, Department of M.Tech CSE, Nitte Meenakshi Institute of Technology Yelahanka, Bangalore-560 064, India*

---***---

Abstract— *Scope of properly identifying the leaves very vast. The properly identifying medicinal leaves for various cures, identifying poisonous plants using leaves, determining the usage of the plant using detected leaves are some of the possible usages of leaf identification. The aim is to build a methodology using various feature extraction techniques to extract features, clustering algorithm to cluster the features and decision trees as a classifier. All these methodologies put together to form an effective method to efficiently recognize the unknown leaf image using trained model. Feature extraction techniques like SIFT and SURF which are robust and provide matching in spite of the change in intensity, size or rotation of the object in the images. Effective corner points are chosen from the image from which magnitude and orientation of surrounding are used to build descriptor that is the vector of feature for each corner points. For Clustering the data, various partitional, hierarchical, density based methods are used to cluster the data which cluster the data with respect to inter-connectivity, similarity, closeness, etc. The clusters data is used to build the decision tree like C4.5 and CART which uses entropy and Gini index as the splitting criteria. The unknown image features are used to traverse the decision tree of the closest cluster to yield the matching image output from the training set.*

***Keywords—SIFT, SURF, ORB, Chameleon Clustering, Decision Tree Classifier*.**

I. INTRODUCTION

Various types of plants and trees having different types of leaves with a different shape, sizes, patterns, textures, etc play an various different role in human life. The various plants, trees or leaves can be identified effectively using their respective leaf. Due to a millions variety of leaves and plants available in the universe, it is not possible for everyone to know each and every one. Since a lot of them are very useful and many are useless, it is very useful to correctly identify the plants using their leaves. Hence this paper provides a mechanism for correctly training and identifying commonly used leaves.

Object detection is the process of finding instances of real-world objects such as faces, vehicles, intruder, etc in images or videos. Object detection algorithms typically use extracted features and learning algorithms to recognize category or label of the object. It is commonly used in applications such as image retrieval, security, surveillance, and automated vehicle parking systems.

For feature extraction, various appearance-based and feature based methods have been developed to overcome various factors like illuminance, rotation, noise, scaling,etc. Appearance based methods are relatively older, less accurate and susceptible to various factors. It includes color based methods, Edge matching, gradient matching, geometrical shapes matching, etc. But evolving feature based methods are very accurate, fast and robust. Methods like SIFT, SURF, FAST, BRIEF, ORB, etc are the methods which are most popular and vastly used nowadays for object detection.

Clustering is one of the important aspect vastly used in data mining and Machine Learning. Over the years, the various algorithms have been evolved for clustering. Partitioning based algorithms like K-means, PAM, CLARA, etc., Hierarchical based methods like BIRCH, ROCK, CURE, etc., Density based methods like DBSCAN are commonly used popular clustering methods. In this paper, CURE clustering has been used to cluster the features since CURE is capable of handling large dataset, is relatively faster and can find the clusters of various shapes and sizes.

Decision Trees is the one the mostly used classifier in the Machine learning. There are various types of decision trees like ID3, C4.5, CART, etc which are used for various different purpose and data which are continuously evolving. The decision trees are different due to splitting criteria like entropy, Gini index, etc, various pruning methods and other capabilities.

In this paper, SIFT or SURF is used for feature extraction of leaf images which are then clustered using CURE clustering algorithm whose decision tree is built using C4.5 or CART methods. The new leaf image is identified by traversing the existing decision tree of the cluster for which query features are having least distance.

II. LITERATURE REVIEW

A. SIFT CORNER DETECTOR

David Lowe presented a new methodology to obtain distinctive invariant features from images in 2002 called Scale Invariant Feature Transform (SIFT) [5]. These features are invariant to translations, rotations, and scaling of the image, and provides robust matching across perspective transformations, distortions, change in 3D viewpoints, noise, and illumination variations. They are well localized in both the spatial and frequency domains, reducing the probability of disruption by occlusion, clutter, or noise. Large numbers of features can be extracted from typical images with efficient algorithms. In addition, the features are highly distinctive, which allows a single feature to be correctly matched with high probability against a large database of features, providing a basis for object and scene recognition. The cost of extracting these features is minimized by taking a cascade filtering approach, in which the more expensive operations are applied only at locations that pass an initial test. Following are the major stages of computation used to generate the set of image features:

a. Scale-space extrema detection

The first stage of computation searches over all scales and image locations. It is implemented efficiently by using a difference-of-Gaussian function to identify potential interest points that are invariant to scale and orientation. Fig 1 depicts the Images in a different scales, and compotation of DoG.

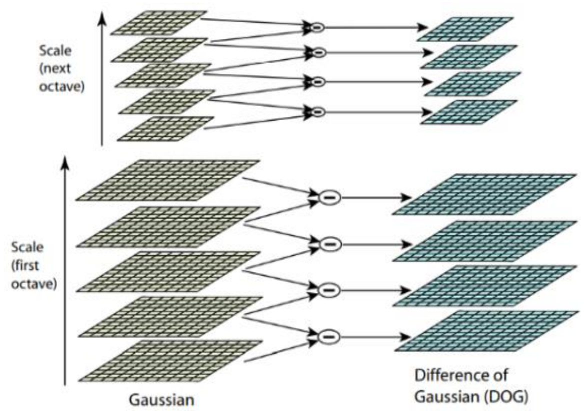

Fig.1 Images in a different scales, and compotation of DoG from [5]

b. Accurate Keypoint localization

At each candidate location, a detailed model is fit to determine location and scale. Keypoints are selected based on measures of their stability.

Once a keypoint candidate has been found by comparing a pixel to its neighbors, the next step is to perform a detailed fit to the nearby data for location, scale, and the ratio of principal curvatures. This information allows points to be rejected that have low contrast (and are therefore sensitive to noise) or are poorly localized along an edge.

c. Orientation assignment

One or more orientations are assigned to each keypoint location based on local image gradient directions to achieve invariance to image rotation. A neighborhood is taken around the keypoint location depending on the scale, and the gradient magnitude and direction are calculated in that region. An orientation histogram with 36 bins covering 360 degrees is created. (It is weighted by gradient magnitude and Gaussian-weighted circular window with sigma equal to 1.5 times the scale of keypoint. The highest peak in the histogram is taken and any peak above 80% of it is also considered to calculate the orientation. It creates keypoints with same location and scale, but different directions. It contributes to the stability of matching.

d. Keypoint descriptor

Here, keypoint descriptor for each keypoint is created. A 16x16 neighborhood around the keypoint is taken which is divided into 16 sub-blocks of 4x4 size. For each sub-block, 8 bin orientation histograms are created. So, a total of 128 bin values are available. It is represented as a vector to form keypoint descriptor. In addition, several methods are applied to achieve robustness against illumination changes, rotation etc.

B. SURF CORNER DETECTORS

Speeded up robust features(SURF) algorithm [3] is the feature point extraction algorithm purposed by Bay H,TuytelaarsT,Gool L V in 2006. This algorithm is similar to SIFT algorithm. SURF is a scale and rotation-invariant interest point detector and a descriptor which is computationally very fast. It uses Integral images to improve the speed. The key points are detected by using a Fast-Hessian matrix. The descriptor describes a distribution of Haar-wavelet responses within the interest point neighborhood. The SURF detector algorithm can be summarized by following steps:

a. Fast-Hessian detector (interest point detection)

The performance in SURF can be ascribed to the use of an intermediate image representation is known as the Integral Image. The integral image is computed rapidly from an input image and is used to speed up the calculation of any upright rectangular area.

The SURF detector is based on the determinant of the Hessian matrix. The determinant responses are normalized to scale at this stage of the algorithm. The higher scale is used the more pixels pass into the kernel and the largest determinant response is received. This avoids the probability that many features will be found in higher scale when non-maximal suppression is used. The non-maximal suppression, calculated at the fourth stage, is based on finding the maximum determinant value within 26 nearest neighbors in lower, present and upper scale. Afterwards, this value is filtered with a predefined threshold to reserve strongest interest point only.

b. Descriptor orientation assignment

At this stage, the Haar wavelets of size 4σ in x and y directions are computed for the orientation assignment. Wavelets are calculated for pixels that are located within a radius of 6σ around the interest points. The dominant orientation is evaluated by the sum of vertical and horizontal responses. The descriptor is calculated at the last stage, using Haar wavelets in square 20σ size area centered at the interest point and oriented along the dominant direction.

C. CURE CLUSTERING

(Clustering Using Representatives) CURE [1] is a bottom-up hierarchical clustering algorithm, but instead of using a centroid-based approach or an all-points approach it employs a method that is based on choosing a well-formed group of points to identify the distance between clusters. Fig.2 shows the flow of the CURE clustering Algorithm.

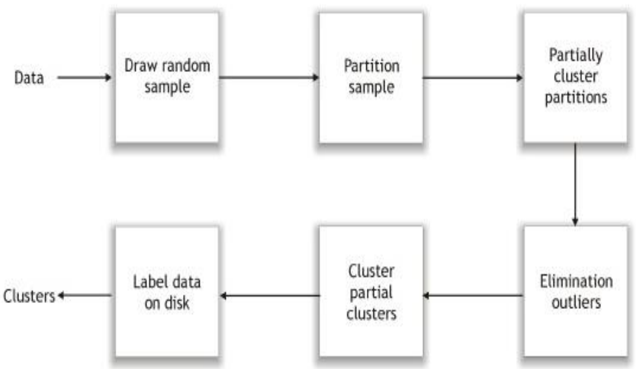

Fig. 2 CURE Architecture

Instead of using single centroid or all points to represent a cluster, a fixed number of representative points in a cluster are chosen. The representative points are generated by randomly selecting the scattered points from all points in the cluster, then shrinking these scattered points toward the mean of the cluster. CURE overcomes the problem of favoring clusters with a spherical shape and similar sizes and is more robust with respect to outliers.

The major steps of the CURE algorithm are as follows:

1. Pick up random sample of points from dataset

2. Cluster sample points hierarchically to create the initial clusters

3. Pick representative Points

- For each cluster pick k - representative point dispersed across clusters
- Move each representative point a fixed fraction towards the centroid of cluster

4. Labeling data

- Rescan the whole dataset and visit each point in dataset and place in the closest cluster
- The closest cluster to the point p is determined by comparing representative point closest to the point p.

The advantages of CURE are as follows:

• CURE can discover clusters with interesting shapes

•CURE is not very sensitive to outliers and the algorithm filters them well

•Sampling and partitioning reduce input size without sacrificing cluster quality

• Execution time of CURE is low compared to most of the existing clustering algorithms

D. C4.5 DECISION TREE

Decision trees [16] are a very effective method of supervised learning. It aims is the partition of a dataset into groups as homogeneous as possible in terms of the variable to be predicted.

C4.5 [9] is a decision tree technique for inducing classification rules in the form of decision trees from a set of given instances. C4.5 is a software extension of the basic ID3 algorithm[8] designed by Quinlan (1979, 1983). Being a supervised learning algorithm, it requires an arrangement of preparing cases and every illustration can be viewed as a pair: input object and the desired output value (class). The classifier used by C4.5 is a decision tree and this tree is built from root to leaves by choosing a simpler solution and a certain splitting criterion.

Entropy and Information gain is the criteria which are used in the C4.5 which is same for discrete variables as in ID3 algorithm but additionally, C4.5 supports continuous or numeric data which splits node using a threshold of the data in selected attribute yielding binary splits.

The major improvement of C4.5 decision tree over ID3 are:

(i) Accepts both continuous and discrete features.

(ii) Can handle missing data.

(iii) Tree pruning using various methods.

(iv) Different weights can be applied the features that comprise the training data.

E. CART Decision Tree

Classification And Regression Trees (CART) [17] is developed by Bierman, Friedman, Olsten, Stone in early 80's.CART is a nonparametric decision tree learning technique can be used for building both Classification and Regression Decision Trees. The impurity (or purity) measure used in building a decision tree in CART is Gini Index. The decision tree built by CART algorithm is always a binary decision tree (each node will have only two child nodes).

The main advantages are (i) CART handles missing values automatically Using "surrogate splits" (ii) Invariant to monotonic transformations of predictive variable (iii) Not sensitive to outliers in predictive variables.

III. Proposed Methodology

Our Proposed method consists of the multi-step methodology which consists of various feature extraction techniques along with newer clustering method and various types of decision trees for classification. Various machine learning techniques and methodologieshave been used to build arobust and effective framework for various type object detection like commonly used leaves. The proposed method also aims to do acomparative analysis of SIFT and SURF feature extractions techniques along with the C4.5 and CART decision tree for classification in our given context.Proposed method is further divided into two parts: Identification and Training.

A. Training Phase

As the name suggests, training of various types of commonly used leaves dataset is used to train our machine learning framework. Fig. 3 depicts the major steps involved in training phase:

- Feature Extraction: For the interest point or features of the training leaf images SIFT and SURF algorithms is used once at a time and perform a comparative analysis with respect to performance and accuracy.
- Clustering: Features extracted is merged to form a training dataset to perform clustering as shown in Figure 3. In this paper, CURE a hierarchical clustering method is used to partition the training dataset into specified k number of clusters.
- Decision Tree: At last, the decision tree is built for each individual cluster formed. Here also two different decision tree methods C4.5 and CART is used for relative comparison between them.

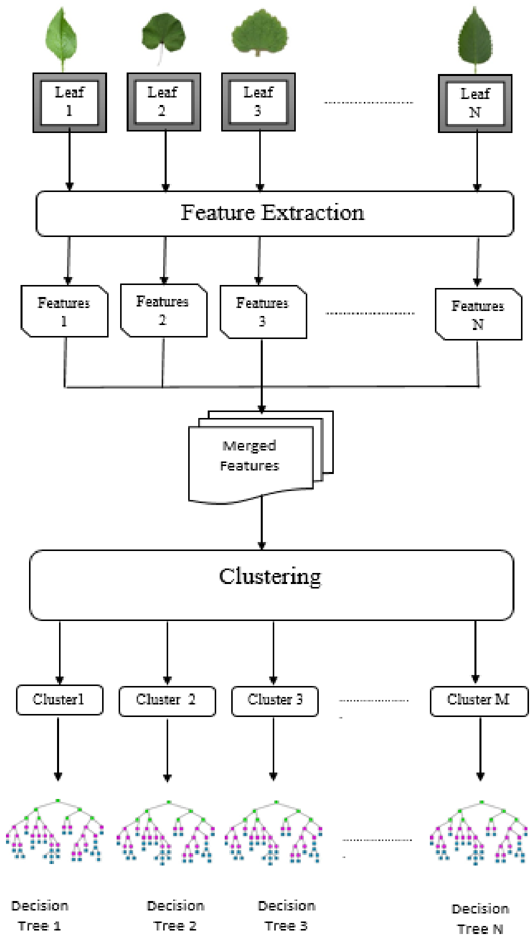

Fig. 3 Flow diagram of Training Phase

B. Identification Phase

After the completion of training, identification phase is used to detect unknown leaf using our trained model .Fig 4. Depicts the steps involved in Training Phase:

- Initially, the features are extracted from the query image using SIFT or SURF
- Each feature keypoint is then compared with the centroid of clusters (i.e mean of cluster points) formed during training period using Euclidian distance
- Then, the respective decision tree of cluster matched is traversed to find the matching label of that keypoint
- Finally, all the output from each keypoint is combined and then voting is used to determine the matching leaf.

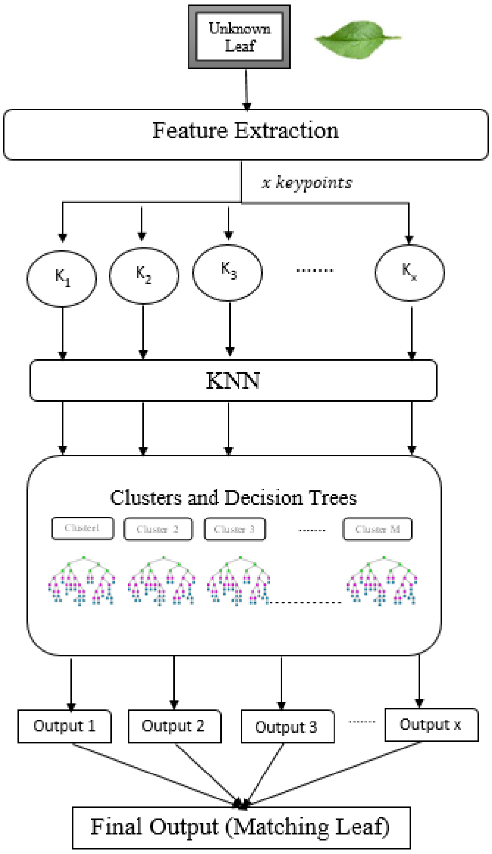

Fig. 4 Flow diagram of Testing Phase

IV. EXPERIMENT WORK, COMPARISON AND ANALYSIS

In the current work, set of 10, 20, 30 and 40 medical Leafs with a different structure, shape, and sizes are taken as a training dataset for given model. Also, the number of query images 5,8,12 and 15 respectively for given dataset is chosen from training images which are scaled and rotated. The accuracy of the proposed work is calculated using a number of query image correctly identified and percentage of correct matching of the keypoints.

In addition, for the purpose of comparison, the multiple numbers of experiments are carried out varying SIFT or SURF for feature extraction with reduced keypoints along with different decision trees i.e. C4.5 and CART used for classification. Fig 5 shows the training leaf dataset.

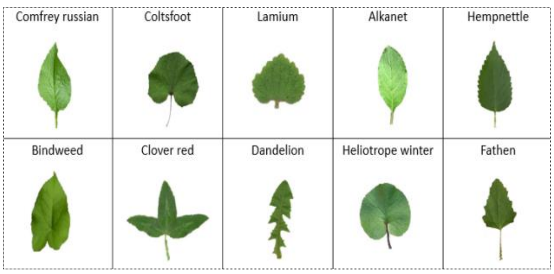

Fig. 5 Training Leaf Dataset

The results of the experiments carried out are as follows:

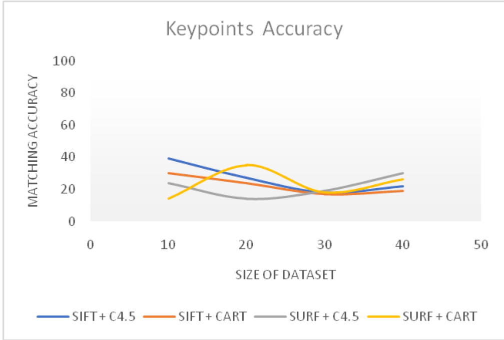

Fig. 6 Comparison of Keypoint Accuracy of SIFT and SURF

From the number of experiments and observations the performance of SIFT and SURF features along with decision tree with entropy as splitting criteria has been found better compared to others with respect to matching accuracy as shown in the graph above. Although the percentage of keypoints been matched for a selected sample leaf image was found almost same for all others.

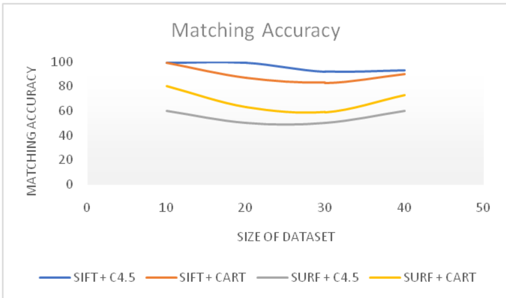

Fig.7 Matching Accuracy of SIFT and SURF

From the number of experiments and observations the performance of SIFT and SURF features along with decision tree with entropy as splitting criteria has been found better compared to others with respect to matching accuracy as shown in the graph above. Although the percentage of keypoints been matched for a selected sample leaf image was found almost same for all others.

V. CONCLUSION AND FUTURE WORK

Using the proposed work in this paper we have successfully compared SIFT and SURF feature descriptors along with Entropy and Gini index as decision tree criteria features for the leaf image matching in given scenario. As per current comparative analysis using proposed method, SIFT for the feature extraction and C4.5 has been found suitable for the best possible number of matches. The CURE clustering algorithm has been successfully able to cluster feature data with various dimensions.

The use of current proposed method gives us efficient mechanism to match leaf images in training dataset. But there is various new feature extraction techniques other than SIFT and SURF that can perform better, various other clustering techniques and decision trees yet to be tested in our proposed method.

ACKNOWLEDGMENTS

Our sincere thanks goes to the Vision Group of Science and Technology (VGST), Karnataka to acknowledge our research and provide us the financial support to carry out the research at NMIT. The authors express their sincere gratitude to Ms. Shilpa Ankalaki for helping author during preparation of the manuscript of the paper.

Finally our sincere gratitude goes to Prof. N R Shetty, Director NMIT and Dr. H C Nagaraj, Principal NMIT for providing the infrastructure support and whole hearted encouragement to carry out the research at NMIT.

REFERENCE:

1. Guha, Sudipto, Rajeev Rastogi, and Kyuseok Shim. "CURE: an efficient clustering algorithm for large databases." ACM Sigmod Record. Vol. 27. No. 2. ACM, 1998.
2. Xu, Rui, and Donald Wunsch. "Survey of clustering algorithms." IEEE Transactions on neural networks 16.3 (2005): 645-678.
3. Bay, Herbert, Tinne Tuytelaars, and Luc Van Gool. "Surf: Speeded up robust features." Computer vision–ECCV 2006 (2006): 404-417
4. Bay, Herbert, et al. "Speeded-up robust features (SURF)." Computer vision and image understanding 110.3 (2008): 346-359.
5. Lowe, D.: Distinctive image features from scale-invariant keypoints, cascade filtering approach. IJCV 60, 91–110 (2004)
6. He, Wei, et al. "Surf tracking." Computer Vision, 2009 IEEE 12th International Conference on. IEEE, 2009.
7. Safavian, S. Rasoul, and David Landgrebe. "A survey of decision tree classifier methodology." IEEE transactions on systems, man, and cybernetics21.3 (1991): 660-674.
8. Quinlan, J. Ross. "Induction of decision trees." Machine learning 1.1 (1986): 81-106.
9. Quinlan, J. Ross. C4. 5: programs for machine learning. Elsevier, 2014.
10. Quinlan, J. Ross. "Improved use of continuous attributes in C4. 5." Journal of artificial intelligence research 4 (1996): 77-90.
11. Quinlan, J. Ross. "Bagging, boosting, and C4. 5." AAAI/IAAI, Vol. 1. 1996.
12. Kohavi, Ronny, and J. Ross Quinlan. "Data mining tasks and methods: Classification: decision-tree discovery." Handbook of data mining and knowledge discovery. Oxford University Press, Inc., 2002.
13. Hui, Dong, and Han Dian Yuan. "Research of image matching algorithm based on SURF fea-tures." Computer Science and Information Processing (CSIP), 2012 International Conference on. IEEE, 2012.
14. Hssina, Badr, et al. "A comparative study of decision tree ID3 and C4. 5." International Journal of Advanced Computer Science and Applications 4.2 (2014): 13-19.
15. Breiman, Leo. "Bagging predictors." Machine learning 24.2 (1996): 123-140.
16. Bradski, Gary, and Adrian Kaehler. Learning OpenCV: Computer vision with the OpenCV library. " O'Reilly Media, Inc.", 2008